Listing of the Claims:

E1

4. (Previously presented) A method for representing a given range of numbers with an optimized set of entries utilizing wildcards, the given range having a beginning number and an ending number, wherein the given range includes a first sub-range, a second sub-range, a third sub-range, and a fourth sub-range, the first sub-range having lower numbers than the second sub-range, which has lower numbers than the third sub-range, which has lower numbers than the fourth sub-range, the method comprising:

representing all numbers within the first sub-range as entries within the optimized set; and

representing and optimizing the second, third, and fourth sub-ranges as a plurality of entries using wildcards within the optimized set, wherein the optimizing only includes the given range of numbers.

38. (Previously presented) A method for representing a range of numbers by an optimized set of hierarchically ordered sub-ranges using wildcard entries, the range having a lowest value range number and a highest value range number, wherein each of the sub-ranges includes a lowest value sub-range number and a highest value sub-range number, the method comprising:

(a) generating a set of sub-ranges from the range; and

(b) optimizing at least one of the sub-ranges, said optimizing (b) including at least:

(c) determining a difference position between a lowest value sub-range number and a highest value sub-range number indicating a maximal degree of optimization of the sub-range, and

(d) optimizing the sub-range based upon the difference position.

39. (Canceled) A method as recited in claim 38 wherein said optimizing (b) further includes at least:

(e) determining a counting value and a magnitude value wherein the magnitude value indicates a number of wildcards used to optimize the sub-range;

(f) truncating the highest value sub-range number based upon the difference position to form a limiting value;

13

(g) adding a wildcard entry to the optimized set based upon the counting value and the magnitude value;

(h) incrementing the counting value; and

(i) repeating (g) and (h) until the counting value equals the limiting value or the counting value is divisible by ten.

40. (Previously Presented) A method as recited in claim 38, wherein the determining (c) comprises:

comparing the lowest value sub-range number and the highest value sub-range number from a most significant digit position to a least significant digit position of each, wherein the difference position is a first position where the comparing is different.

41. (Previously Presented) A method as recited in claim 40, further comprising:

for the lowest value sub-range number, determining a number of contiguous zero digits from the least significant digit position;

dropping off the number of contiguous zero digits from the lowest value sub-range number to form the counting value; and

setting the magnitude value as the number of dropped contiguous zero digits.

42. (Canceled) A method as recited in claim 39, wherein when the counting value is divisible by ten then,

forming a new counting value by dropping zero digits off of the counting value starting at a least significant digit position; and

incrementing the magnitude value by the number of dropped zero digits.

43. (Canceled) A method as recited in claim 39, wherein when the counting value is equal to the limiting value, then the optimizing (d) ends for the sub-range.

44. (Previously Presented) A method as recited in claim 38, wherein for a second sub-range, a second sub-range lowest value number is one more than a first sub-range highest value number.

45. (Previously Presented) A method as recited in claim 44 wherein the first sub-range is formed of the range of numbers starting at the lowest value range number up to but not including a first range number divisible by ten.

46. (Previously Presented) A method as recited in claim 45, wherein when the lowest range number is divisible by an n^{th} power of ten, the first n^{th} sub-ranges each have zero entries.

47. (Previously Presented) A method as recited in claim 38, wherein each of the range of numbers represents a telephone number.

48. (Previously Presented) A method as recited in claim 38, wherein each of the range of numbers represents a router address.

49. (Previously Presented) A computing system for representing a range of numbers by an optimized set of hierarchically ordered sub-ranges using wildcard entries, the range having a lowest value range number and a highest value range number, wherein each of the sub-ranges includes a lowest value sub-range number and a highest value sub-range number, comprising:

a memory; and

a processor coupled to the memory arranged to execute programming code for

(a) generating a set of sub-ranges from the range; and

(b) optimizing at least one of the sub-ranges, said optimizing (b) including at least:

E1

(c) determining a difference position between a lowest value sub-range number and a highest value sub-range number indicating a maximal degree of optimization of the sub-range, and

(d) optimizing the sub-range based upon the difference position.

50. (Canceled) A computing system as recited in claim 49, wherein said optimizing (b) further includes at least:

(e) determining a counting value and a magnitude value wherein the magnitude value indicates a number of wildcards used to optimize the sub-range;

(f) truncating the highest value sub-range number based upon the difference position to form a limiting value;

(g) adding a wildcard entry to the optimized set based upon the counting value and the magnitude value;

(h) incrementing the counting value; and

(i) repeating (g) and (h) until the counting value equals the limiting value or the counting value is divisible by ten.

51. (Previously Presented) A computing system as recited in claim 49, wherein the determining (c) comprises:

comparing the lowest value sub-range number and the highest value sub-range number from a most significant digit position to a least significant digit position of each, wherein the difference position is a first position where the comparing is different.

52. (Previously Presented) A computing system as recited in claim 51, further comprising:

for the lowest value sub-range number, determining a number of contiguous zero digits from the least significant digit position;

dropping off the number of contiguous zero digits from the lowest value sub-range number to form the counting value; and

setting the magnitude value as the number of dropped contiguous zero digits.

E1

53. (Canceled) A computing system as recited in claim 50, wherein when the counting value is divisible by ten then,

forming a new counting value by dropping zero digits off of the counting value starting at a least significant digit position; and

incrementing the magnitude value by the number of dropped zero digits, wherein when the counting value is equal to the limiting value, then the optimizing (d) ends for the sub-range.

54. (Previously Presented) A computer program product for representing a range of numbers by an optimized set of hierarchically ordered sub-ranges using wildcard entries, the range having a lowest value range number and a highest value range number, wherein each of the sub-ranges includes a lowest value sub-range number and a highest value sub-range number, comprising:

computer code for generating a set of sub-ranges from the range;

computer code for optimizing at least one of the sub-ranges, said optimizing code including at least:

computer code for determining a difference position between a lowest value sub-range number and a highest value sub-range number indicating a maximal degree of optimization of the sub-range,

computer code for optimizing the sub-range based upon the difference position; and

computer readable medium for storing the computer code.

55. (Canceled) A computer program product as recited in claim 54, wherein said computer code for optimizing further includes at least:

computer code for determining a counting value and a magnitude value wherein the magnitude value indicates a number of wildcards used to optimize the sub-range;

computer code for truncating the highest value sub-range number based upon the difference position to form a limiting value;

computer code for adding a wildcard entry to the optimized set based upon the counting value and the magnitude value;

E1

computer code for incrementing the counting value; and

computer code for repeating adding the wildcard entry and the incrementing the counting value until the counting value equals the limiting value or the counting value is divisible by ten.

56. (Previously Presented) A computer program product as recited in claim 54, wherein the code for determining a difference position comprises:

computer code for comparing the lowest value sub-range number and the highest value sub-range number from a most significant digit position to a least significant digit position of each, wherein the difference position is a first position where the comparing is different.

57. (Previously Presented) A computer program product as recited in claim 56, further comprising:

for the lowest value sub-range number, computer code for determining a number of contiguous zero digits from the least significant digit position;

computer code for dropping off the number of contiguous zero digits from the lowest value sub-range number to form the counting value; and

computer code for setting the magnitude value as the number of dropped contiguous zero digits.

58. (Previously Presented) A computer program product as recited in claim 54, wherein when the counting value is divisible by ten then the set of programming instructions further includes:

computer code for forming a new counting value by dropping zero digits off of the counting value starting at a least significant digit position; and

computer code for incrementing the magnitude value by the number of dropped zero digits, wherein when the counting value is equal to the limiting value, then the optimizing (d) ends for the sub-range.

E1

59. (Previously Presented) A router for representing a range of numbers by an optimized set of hierarchically ordered sub-ranges using wildcard entries, the range having a lowest value range number and a highest value range number, wherein each of the sub-ranges includes a lowest value sub-range number and a highest value sub-range number, comprising:

a memory; and

a processor coupled to the memory arranged to execute the computer program product as recited in claim 54.

60. (Previously Presented) A apparatus for representing a range of numbers by an optimized set of hierarchically ordered sub-ranges using wildcard entries, the range having a lowest value range number and a highest value range number, wherein each of the sub-ranges includes a lowest value sub-range number and a highest value sub-range number, the method comprising:

means for generating a set of sub-ranges from the range; and

means for optimizing at least one of the sub-ranges, said optimizing means including at least:

means for determining a difference position between a lowest value sub-range number and a highest value sub-range number indicating a maximal degree of optimization of the sub-range, and

means for optimizing the sub-range based upon the difference position.

ALLOWED CLAIMS

61. (Previously Presented) A method for representing a range of numbers by an optimized set of hierarchically ordered sub-ranges using wildcard entries, the range having a lowest value range number and a highest value range number, wherein each of the sub-ranges includes a lowest value sub-range number and a highest value sub-range number, the method comprising:

(a) generating a set of sub-ranges from the range; and

(b) optimizing at least one of the sub-ranges, said optimizing (b) including at least:

(c) determining a difference position between a lowest value sub-range number and a highest value sub-range number indicating a maximal degree of optimization of the sub-range;

E1

(d) optimizing the sub-range based upon the difference position;

(e) determining a counting value and a magnitude value wherein the magnitude value indicates a number of wildcards used to optimize the sub-range;

(f) truncating the highest value sub-range number based upon the difference position to form a limiting value;

(g) adding a wildcard entry to the optimized set based upon the counting value and the magnitude value;

(h) incrementing the counting value; and

(i) repeating (g) and (h) until the counting value equals the limiting value or the counting value is divisible by ten.

62. (Previously Presented) A method as recited in claim 61, wherein the determining (c) comprises:

comparing the lowest value sub-range number and the highest value sub-range number from a most significant digit position to a least significant digit position of each, wherein the difference position is a first position where the comparing is different.

63. (Previously Presented) A method as recited in claim 62, further comprising:

for the lowest value sub-range number, determining a number of contiguous zero digits from the least significant digit position;

dropping off the number of contiguous zero digits from the lowest value sub-range number to form the counting value; and

setting the magnitude value as the number of dropped contiguous zero digits.

64. (Previously Presented) A method as recited in claim 61, wherein when the counting value is divisible by ten then,

13

forming a new counting value by dropping zero digits off of the counting value starting at a least significant digit position; and

incrementing the magnitude value by the number of dropped zero digits.

65. (Previously Presented) A method as recited in claim 61, wherein when the counting value is equal to the limiting value, then the optimizing (d) ends for the sub-range.

66. (Previously Presented) A method as recited in claim 61, wherein for a second sub-range, a second sub-range lowest value number is one more than a first sub-range highest value number.

67. (Previously Presented) A method as recited in claim 66 wherein the first sub-range is formed of the range of numbers starting at the lowest value range number up to but not including a first range number divisible by ten.

68. (Previously Presented) A method as recited in claim 67, wherein when the lowest range number is divisible by an n^{th} power of ten, the first n^{th} sub-ranges each have zero entries.

69. (Previously Presented) A method as recited in claim 61, wherein each of the range of numbers represents a telephone number.

70. (Previously Presented) A method as recited in claim 61, wherein each of the range of numbers represents a router address.

71. (Previously Presented) A computing system for representing a range of numbers by an optimized set of hierarchically ordered sub-ranges using wildcard entries, the range having a lowest value range number and a highest value range number, wherein each of the sub-ranges includes a lowest value sub-range number and a highest value sub-range number, comprising:

a memory; and

a processor coupled to the memory arranged to execute programming code for

(a) generating a set of sub-ranges from the range; and

(b) optimizing at least one of the sub-ranges, said optimizing (b) including at least:

(c) determining a difference position between a lowest value sub-range number and a highest value sub-range number indicating a maximal degree of optimization of the sub-range;

(d) optimizing the sub-range based upon the difference position.

(e) determining a counting value and a magnitude value wherein the magnitude value indicates a number of wildcards used to optimize the sub-range;

(f) truncating the highest value sub-range number based upon the difference position to form a limiting value;

(g) adding a wildcard entry to the optimized set based upon the counting value and the magnitude value;

(h) incrementing the counting value; and

(i) repeating (g) and (h) until the counting value equals the limiting value or the counting value is divisible by ten.

72. (Previously Presented) A computing system as recited in claim 71, wherein the determining (c) comprises:

comparing the lowest value sub-range number and the highest value sub-range number from a most significant digit position to a least significant digit position of each, wherein the difference position is a first position where the comparing is different.

73. (Previously Presented) A computing system as recited in claim 72, further comprising:

for the lowest value sub-range number, determining a number of contiguous zero digits from the least significant digit position;

dropping off the number of contiguous zero digits from the lowest value sub-range number to form the counting value; and

setting the magnitude value as the number of dropped contiguous zero digits.

61

74. (Previously Presented) A computing system as recited in claim 71, wherein when the counting value is divisible by ten then,

forming a new counting value by dropping zero digits off of the counting value starting at a least significant digit position; and

incrementing the magnitude value by the number of dropped zero digits, wherein when the counting value is equal to the limiting value, then the optimizing (d) ends for the sub-range.

75. (Previously Presented) A computer program product for representing a range of numbers by an optimized set of hierarchically ordered sub-ranges using wildcard entries, the range having a lowest value range number and a highest value range number, wherein each of the sub-ranges includes a lowest value sub-range number and a highest value sub-range number, comprising:

computer code for generating a set of sub-ranges from the range;

computer code for optimizing at least one of the sub-ranges, said optimizing code including at least:

computer code for determining a difference position between a lowest value sub-range number and a highest value sub-range number indicating a maximal degree of optimization of the sub-range;

computer code for optimizing the sub-range based upon the difference position;

computer code for optimizing further includes at least,

computer code for determining a counting value and a magnitude value wherein the magnitude value indicates a number of wildcards used to optimize the sub-range;

computer code for truncating the highest value sub-range number based upon the difference position to form a limiting value;

computer code for adding a wildcard entry to the optimized set based upon the counting value and the magnitude value;

computer code for incrementing the counting value; and

computer code for repeating adding the wildcard entry and the incrementing the counting value until the counting value equals the limiting value or the counting value is divisible by ten; and

computer readable medium for storing the computer code.

E1

76. (Previously Presented) A computer program product as recited in claim 75, wherein the code for determining a difference position comprises:

computer code for comparing the lowest value sub-range number and the highest value sub-range number from a most significant digit position to a least significant digit position of each, wherein the difference position is a first position where the comparing is different.

77. (Previously Presented) A computer program product as recited in claim 75, further comprising:

for the lowest value sub-range number, computer code for determining a number of contiguous zero digits from the least significant digit position;

computer code for dropping off the number of contiguous zero digits from the lowest value sub-range number to form the counting value; and

computer code for setting the magnitude value as the number of dropped contiguous zero digits.

78. (Previously Presented) A computer program product as recited in claim 77, wherein when the counting value is divisible by ten then the set of programming instructions further includes:

computer code for forming a new counting value by dropping zero digits off of the counting value starting at a least significant digit position; and

computer code for incrementing the magnitude value by the number of dropped zero digits, wherein when the counting value is equal to the limiting value, then the optimizing (d) ends for the sub-range.

79. (Previously Presented) A router for representing a range of numbers by an optimized set of hierarchically ordered sub-ranges using wildcard entries, the range having a lowest value range number and a highest value range number, wherein each of the sub-ranges includes a lowest value sub-range number and a highest value sub-range number, comprising:

a memory; and

a processor coupled to the memory arranged to execute the computer program product as recited in claim 75.

80. (Previously Presented) A apparatus for representing a range of numbers by an optimized set of hierarchically ordered sub-ranges using wildcard entries, the range having a lowest value range number and a highest value range number, wherein each of the sub-ranges includes a lowest value sub-range number and a highest value sub-range number, the method comprising:

means for generating a set of sub-ranges from the range;

means for optimizing at least one of the sub-ranges, said optimizing means including at least,

means for determining a difference position between a lowest value sub-range number and a highest value sub-range number indicating a maximal degree of optimization of the sub-range;

means for optimizing the sub-range based upon the difference position(c) determining a difference position between a lowest value sub-range number and a highest value sub-range number indicating a maximal degree of optimization of the sub-range;

means for optimizing the sub-range based upon the difference position.

means for determining a counting value and a magnitude value wherein the magnitude value indicates a number of wildcards used to optimize the sub-range;

means for truncating the highest value sub-range number based upon the difference position to form a limiting value;

means for adding a wildcard entry to the optimized set based upon the counting value and the magnitude value;

means for incrementing the counting value; and

means for repeating adding the wildcard entry and the incrementing the counting value until the counting value equals the limiting value or the counting value is divisible by ten.